PATENT
98086PCTUS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Mittenthal, Lothrop	)	**Examiner:**	LaForgia, Christian A.
		)		
Serial No.:	09/762,555	)	**Art Unit:**	2131
		)		
Filing Date:	4/10/2001	)	**Attorney Docket No.**	98086PCTUS

Title: DETERMINISTICALLY GENERATING BLOCK SUBSTITUTION TABLES WHICH MEET A GIVEN STANDARD OF NONLINEARITY

APPEAL BRIEF OF APPLICANT LOTHROP MITTENTHAL

Christopher G. Wolfe
Reg. No. 56,264

KIRKPATRICK & LOCKHART NICHOLSON
GRAHAM LLP
Henry W. Oliver Building
535 Smithfield Street
Pittsburgh, PA 15222

Ph. (412) 355-6798
Fax (412) 355-6501
cwolfe@klng.com

TABLE OF CONTENTS

Applicant for the above-identified patent application, Lothrop Mittenthal, submits this appeal brief in accordance with the provisions of 37 C.F.R. 41.37 in response to (i) the Office Action dated May 18, 2005, (ii) the Notice of Appeal filed August 18, 2005, and (iii) the Notice of Abandonment mailed on March 29, 2006. A check for the appropriate fees under 37 C.F.R. 41.20(b)(2) is enclosed. Applicant is also concurrently filing a "Petition for Revival of an Application For Patent Abandoned Unintentionally Under 37 C.F.R. 1.137(b)" and check for the appropriate fees under that section. Nonetheless, the Commissioner is hereby authorized to charge Account No. 11-1110 for any fees necessary for consideration of this brief or the concurrently filed petition.

I. **REAL PARTY IN INTEREST**

The real party in interest is Teledyne Technologies Incorporated, which is the owner of the instant application by virtue of an assignment from the inventor recorded at Reel 011690, Frame 0409.

II. RELATED APPEALS AND INTERFERENCES

A Notice of Appeal was filed in the instant application on August 18, 2005. Applicant unintentionally failed to file an appeal brief, and a Notice of Abandonment was mailed by the Patent and Trademark Office (PTO) on March 29, 2006.

Applicant is not aware of any other appeals or interferences that will directly affect or be directly affected by or have a bearing on the decision of the Board of Patent Appeals and Interferences ("Board") in the present case.

III. STATUS OF THE CLAIMS

In the non-final Office Action mailed on May 18, 2005 (hereinafter "the Office Action"), pending claims 1, 2, 4 and 7-22 are rejected under 35 U.S.C. section 112, paragraph 2 for failing to point out and distinctly claim the subject matter which Applicant regards as his invention. Also in the Office Action, claims 1, 2, 4 and 7-22 are rejected under 35 U.S.C. section 103(a) as being unpatentable over U.S. Patent No. 6,182,216 to Luyster in view of U.S. Patent No. 5,317,639 to Mittenthal.

Claims 3 and 5-6 were cancelled without prejudice or disclaimer by amendment in Applicant's "Response To Office Action" submitted on August 4, 2004.

Applicant proposes below to cancel claims 11, 14, 15, 18-19 and 22 without prejudice or disclaimer.

Accordingly, upon entry of Applicant's proposed amendment, each of claims 1-2, 4, 7-10, 12-13, 16-17 and 20-21 will stand rejected and subject to appeal. The text of claims 1-2, 4, 7-10, 12-13, 16-17 and 20-21 is set forth in the Claims Appendix beginning at page 22 herein.

IV. STATUS OF AMENDMENTS

Applicant proposes to cancel claims 11, 14, 15, 18-19 and 22 without prejudice or disclaimer. Applicant submits that the cancellation of these claims is in accordance with 37 C.F.R. 41.33(b)(1).

Accordingly, the present appeal is proceeding on claims 1-2, 4, 7-10, 12-13, 16-17 and 20-21 as pending when the Office Action was issued.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

A. Background

Cryptographic block encryption includes a number of methods used to encrypt clear text data by uniquely replacing each *n*-bit word of clear text data with a corresponding *n*-bit word of cipher text data. Common block encryption methods convert clear text data to cipher text data, and visa versa, by utilizing a series of mathematical transformations such as permutation transformations and/or substitution transformations. Permutation transformations include various shifting, mixing and/or reordering of input bits. For example, permutation transformations may include mixing the input bits with a key, mixing a portion of the input bits with a second portion thereof, shifting input bits or blocks thereof, *etc.* Substitution transformations involve applying a block substitution table, or s-box, to input data.

An s-box is basically a look-up table defining a set of output bits for each combination of input bits. For example, the s-box includes an output value for each *n*-bit input. There are various ways to generate s-boxes, including those currently claimed and those taught by U.S. Patent No. 5,317,639 to Mittenthal. It is important to note, however, that there is a clear distinction between generating an s-box or block substitution table, and using an s-box while applying a block encryption cipher.

To apply a block encryption cipher, substitution and permutation transformations are applied to clear text data, usually over multiple rounds. For example, at each round, the output of the previous round may be subjected to various substitution and/or permutation transformations. Often, although not always, each *n*-bit word of clear text data is initially divided into two or more sub-blocks of a predetermined size.[1] *See, e.g.*, Specification at p. 1, ll. 19-21. Various rounds of transformations are then applied to the sub-blocks, with some transformations applied to the sub-blocks individually and some (*e.g.*, mixing) applied together. The result of the rounds of transformations is cipher text data corresponding to the initial clear text data. To decipher the cipher text data, inverse transformations may be applied utilizing the same key and s-box or s-boxes. Only an individual knowing the order of operations, keys, and s-boxes used will be capable of legitimately deciphering the text.

Cryptanalysts possess various tools for breaking ciphers, *i.e.*, deciphering cipher text data without knowing the appropriate keys, s-boxes, and order of operations. *See, e.g.*, Specification at p. 1, ll. 14-29. Block encryption ciphers can be designed, however, to minimize the effectiveness of these tools. For example, ciphers using s-boxes that are highly non-linear are much more difficult to break than other ciphers. Existing methods for generating highly non-linear and maximal non-linear s-boxes are experimental, not deterministic. For example, they involve generating candidate s-boxes then testing the candidate s-boxes to determine their degree of non-linearity.

B. <u>Summary of Subject Matter Defined in Independent Claims</u>

The present application now includes independent claims 1, 16 and 20-21 directed generally to systems, apparatuses and methods for deterministically

[1] This is the methodology used in Feistel-type systems such as the Data Encryption Standard (DES).

generating block substitution tables having high non-linearity, and indeed the theoretical maximal non-linearity. Maximal non-linear block substitution tables generated in this way may be used, for example, as s-boxes in the application of various block encryption ciphers, as described above.

Independent claim 1 of the application states:

> 1. A method of deterministically generating maximal nonlinear block substitution tables for a predetermined block size, comprising:
>
> selecting a first generating function;
>
> selecting a second generating function;
>
> selecting first and second sets of complete linearly independent numbers;
>
> calculating first and second linear orthomorphisms from the generating functions and the sets of linearly independent numbers; and
>
> creating maximal nonlinear block substitution tables by combining the linear orthomorphisms, the block substitution tables for use in encrypting clear text messages.

Independent claims 20 and 21 include similar limitations. According to embodiments of the invention, as recited in claim 1, first and second generating functions are selected. *See, e.g.*, Figure 1A at Ref. Nos. 14, 18, 20, 26 & 28; Figure 2; Specification at p. 7, lines 21-27; p. 13, line 21 - p. 14, line 29. The first and second generating functions are functions that, when applied in conjunction with a set of linearly independent numbers, generate a linear orthomorphism.[2] *See* Specification at p. 6, lines 17-35; p. 20, line 35 – page 21, line 7.

First and second sets of complete linearly independent numbers are selected for use with the first and second generating functions. *See, e.g.*, Figure 1B at Ref. No. 30; Figure 2 at Ref. Nos. 44, 46. Several exemplary methods for generating the sets of complete linearly independent numbers are given at page 20 of the specification, although it will be appreciated that any suitable method may be used. *See* Specification at p. 20, lines 32-34. The first and second generating functions are then applied with the first and second sets of linearly independent numbers, respectively, to generate first and second linear orthomorphisms. *See, e.g.*, Figure 1B at Ref. Nos. 32, 34; Figure 2 at Ref. Nos. 50, 52, 54; Specification at p. 6, lines 17-35; p. 20, line 35 – page 21, line 7. The first and second linear orthomorphisms are combined to generate a maximal non-linear block substitution table. *See, e.g.*, Figure 1C at Ref. Nos. 40, 42; Figure 2 at Ref. No. 56; Specification at p. 21, lines 14-21.

Various steps may be taken to insure that the generated block substitution tables are maximal non-linear. For example, the first generating function may be selected as a primitive polynomial, and the second generating function may be selected as the primitive polynomial that is the complement of the first primitive

[2] An orthomorphism is a function or mapping *f(x)* for which all of the consecutive sums, $x_i + f(x_i)$, are distinct. If a cryptographic mapping is an orthomorphism, then no information about the clear text can be derived from the cipher text. This property of all orthomorphisms is referred to as "lack of mutual information."

polynomial. *See* Figure 1A at Ref. No. 24; Figure 1B at Ref. No. 28; Specification at page 18, line 20 – page 19, line 10.

Independent claim 16 of the application states:

> 16. A computer-implemented method for deterministically generating maximal nonlinear block substitution tables from binary data, comprising:
>
> selecting a first set of a plurality of complete linearly independent numbers from the binary data;
>
> selecting a second set of a plurality of complete linearly independent numbers from the binary data;
>
> generating a plurality of linear orthomorphisms using first and second recursive generating functions and the first and second sets of linearly independent numbers; and
>
> setting the maximal nonlinear substitution tables based on a combination of the linear orthomorphisms, the substitution tables for use in encrypting clear text messages which are in the form of a sequence of binary numbers.

According to embodiments of the invention, as recited in claim 16, first and second sets of a plurality of complete linearly independent numbers are selected from the binary data. *See, e.g.*, Figure 1B at Ref. No. 30; Figure 2 at Ref. Nos. 44, 46. A plurality of linear orthomorphisms are then generated using first and second recursive generating functions and the first and second sets of linearly independent numbers. *See, e.g.*, Figure 1B at Ref. Nos. 32, 34; Figure 2 at Ref. Nos. 50, 52, 54; Specification at p. 6, lines 17-35; p. 20, line 35 – page 21, line 7. The linear orthomorphisms are then combined to set the maximal non-linear substitution tables. *See, e.g.*, Figure 1C at Ref. Nos. 40, 42; Figure 2 at Ref. No. 56; Specification at p. 21, lines 14-21.

Independent claim 20 of the application states:

> 20. A system, comprising:
>
> a communications link;
>
> a first computer in communication with the communications link; and
>
> a second computer in communications with the communications link, the second computer having an ordered set of data and instructions stored thereon which, when executed by the second computer, cause the second computer to perform the steps of:
>
> selecting a first generating function;
>
> selecting a second generating function;
>
> selecting first and second sets of complete linearly independent numbers;
>
> calculating first and second linear orthomorphisms from the generating functions and the sets of linearly independent numbers; and
>
> creating maximal nonlinear block substitution tables by combining the linear orthomorphisms, the block substitution tables for use in encrypting clear text messages.

According to embodiments of the invention, as recited in claim 20, the method of claim 1 above may be implemented in the context of a communication link, a first computer in communications with the communications link, and a second computer in communication with the communications link. *See* Figure 3 at Ref. Nos. 60, 62, 64; Specification at p. 23, lines 8-14.

Independent claim 21 of the application states:

> 21. A computer-readable medium having stored thereon instructions which, when executed by a processor, cause the processor to perform the steps of:
>
> selecting a first generating function;

> selecting a second generating function;
>
> selecting first and second sets of complete linearly independent numbers;
>
> calculating first and second linear orthomorphisms from the generating functions and the sets of linearly independent numbers; and
>
> creating maximal nonlinear block substitution tables by combining the linear orthomorphisms, the block substitution tables for use in encrypting clear text messages.

According to embodiments of the invention, as recited in claim 21, the method of claim 1 is embodied as a computer-readable medium.

C. **Advantages of the Present Invention**

The present invention, as recited in independent claims 1, 16 and 20-21 and explained above, addresses the need for a method of deterministically generating highly non-linear block substitution tables. For example, the maximal non-linear block substitution tables according to claims 1, 16 and 20-21 are generated deterministically, not experimentally. Accordingly, maximal non-linear block substitution tables are generated deterministically (*e.g.*, without the need to exhaustively test the non-linearity of candidate tables).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The Office Action contains two grounds of rejection that are to be reviewed on appeal:

1. Whether claims 1-2, 4, 7-10, 12-13, 16-17 and 20-21 were properly rejected under 35 U.S.C. section 103(a) as being obvious over U.S. Patent No. 6,182,216 to Luyster in view of U.S. Patent No. 5,317,639 to Mittenthal; and

2. Whether claims 1-2, 4, 7-10, 12-13, 16-17 and 20-21 were properly rejected under 35 U.S.C. section 112, second paragraph, for failing to particularly point out and distinctly claim the subject matter which Applicant regards as his invention.

VII. ARGUMENT

A. The Section 103 Rejections Are Improper And Should Be Withdrawn

Claims 1-2, 4, 7-10, 12-13, 16-17 and 20-21 were rejected under 35 U.S.C. section 103(a) as being obvious over U.S. Patent No. 6,182,216 to Luyster ("Luyster") in view of U.S. Patent No. 5,317,639 to Mittenthal ("Mittenthal"). Applicant submits that the claims are patentable over Luyster and Mittenthal, at least because the Office has not supported, and indeed cannot support, the rejections with a *prima facie* case. This is because Luyster and Mittenthal, taken separately or together, fail to teach or suggest all of the limitations of the claims. *See* MPEP § 2142.

1. *Prima Facie* Obviousness

The concept of *prima facie* obviousness determines the burden of going forward with evidence during prosecution. *See id.* If the Office fails to produce a *prima facie* case of obviousness, then an applicant is not even required to submit evidence of non-obviousness. That is, if there is no *prima facie* case of obviousness, then the claims are non-obvious. See *id.*

To establish a *prima facie* case of obviousness, the Office must show that (1) there is, "some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings;" (2) there is a "reasonable expectation of success;" and (3) the prior art reference or references, "teach or suggest all of the claim limitations." *See id.* For the reasons below, Applicant submits that the Office has not provided, and cannot provide a *prima facie* case of obviousness because Luyster and Mittenthal, taken separately or together, fail to teach or suggest, at least, "creating maximal nonlinear block substitution tables by combining the linear orthomorphisms," as recited in claims 1, 20 and 21, or, "setting the maximal nonlinear substitution tables based on a combination of the linear orthomorphisms," as recited in claim 16.

2. Content Of The Prior Art References

a. The Luyster Reference

Luyster teaches methods of multi-round, Feistel-type block encryption.[3] These are methods of applying a block encryption cipher, not methods of generating substitution tables or s-boxes like the presently claimed invention. According to Luyster's methods, a plain text input is divided into two "round segments." In each round, Luyster's methods apply various functions (transformations) to the round segments including, (1) at least one bit-moving function; (2) at least one linear combination function; and (3) at least one non-linear function. *See* Luyster at Abstract, Figures 3, 6, 7, 9, 13 and 14.

Luyster's bit-moving functions involve rotating, shifting, or bit-permuting round segments by a predetermined amount. *See, e.g.*, Luyster at Abstract, Figure 3, Ref. Nos. 58, 70. Luyster's linear combination functions involve combining a round segment with a second input using a linear operator such as, addition, subtraction, exclusive-or (XOR), *etc.* The second input can be another round

[3] Feistel-type methods are those that divide a clear text word into two or more sub-blocks and then perform various transformations on the sub-blocks.

segment, a key or a sub-key. *See, e.g.*, Luyster at Figure 3, Ref. Nos. 56, 60, 62, 72, 74 and 82. Luyster teaches two types of non-linear functions: input-dependent bit-shifting and transformation by block substitution table or s-box. *See* Luyster at Abstract. For example, Luyster's Figure 3 shows an encryption method using input-dependent bit-shifting, *see* Ref Nos. 66, 78, and Figure 7 shows an encryption method using s-box substitutions, *see* Ref Nos. 158, 170.

In addition to its block encryption methods, Luyster also teaches several methods of generating sub-key values from a fixed key. *See* Luyster at Figures 5, 10, 11. Sub-keys are not s-boxes or substitution tables. Instead they are merely sets of values generated from a cipher key that can then be combined with input data in block substitution methods. For example, in Luyster, various sub-key values are linearly combined with round segments during the rounds. *See, e.g.*, Luyster at Figure 3, Ref. Nos. 56, 60, 72. Some of Luyster's methods for generating sub-keys do utilize s-boxes, however, that does not mean that the resulting sub-keys *are* s-boxes. *See* Luyster at Figure 11, Ref. Nos. 256, 266.

Luyster teaches desirable s-box qualities, and alludes to methods of generating s-boxes, but no such methods are explicitly taught. *See* Luyster at col. 47, line 45 – col. 48, line 9. For example, Luyster teaches that it is desirable for an s-box to generate a high minimum number of output-bit differences for any input difference. *See id.* Luyster also teaches that s-boxes generated, or optimized using a permutation method generally have this quality, but that any method producing suitable s-boxes may be used. *See* Luyster at col. 47, line 59 – col. 48, line 3.

b. The Mittenthal Reference

Unlike Luyster, Mittenthal teaches methods of generating s-boxes or substitution tables. Mittenthal teaches generating s-boxes that *are* non-linear orthomorphisms. According to Mittenthal, a linear orthomorphism is generated and then converted into a non-linear orthomorphism.[4] *See* Mittenthal at Abstract. The conversion is accomplished by performing various mathematical steps. *See* Mittenthal at col. 19, ll. 27-28. First, portions of the linear orthomorphism that can be individually non-linearized (corruptible sets of equations) are identified. *See* Mittenthal at col. 19, ll. 47-50. These portions are then non-linearized and assembled into a complete non-linear orthomorphism. *See* Mittenthal at col. 24, ll. 4-17. The non-linear orthomorphism may then be used as an s-box or block substitution table in other encryption methods.

3. Neither Luyster Nor Mittenthal, Taken Individually Or Together, Teach Or Suggest Combining Two Linear Orthomorphisms To Create A Non-Linear Block Substitution Table

a. The Luyster Reference

As discussed above, Luyster is directed to methods of applying block ciphers, and does not teach any method of ***generating*** a non-linear block substitution table. *See* Luyster at Abstract. Luyster does reference a permutation method of generating s-boxes, but does not describe how the permutation method is

[4] Mittenthal does teach that a linear orthomorphism can be found by applying a generating function to a complete linearly independent set of numbers. *See* Mittenthal at col. 48, ll. 44-56.

implemented. *See* Luyster at col. 47, line 59 – col. 48, line 9. In any case, Luyster's permutation method certainly does not involve combining two linear orthomorphisms, or two functions of any kind, to create a non-linear block substitution table as recited by the instant claims.

The Office asserts that Luyster discloses, "combining first and second linear functions to produce a non-linear block substitution table or s-box, as evident by at least the Abstract." *See* Office Action at 5. Applicant submits, however, that this is clearly not the case, as Luyster does not teach combining any functions. Luyster does disclose applying linear combination functions to input data, but this is not combining functions. *See* Luyster at Abstract. Instead, it is simply applying a particular type of function (*i.e.*, a linear combination function) to two input values (*i.e.*, round segments, subkeys, *etc.*) to generate an output value. *See* Luyster at col. 19, ll. 12-38.

b. <u>The Mittenthal Reference</u>

Mittenthal teaches methods of generating block substitution tables, but does not teach any methods that involve combining linear orthomorphisms. Instead, as described above, Mittenthal teaches converting one linear orthomorphism into one non-linear orthomorphism. *See* Mittenthal at col. 19, ll. 27-28. The non-linear orthomorphism may then be used as an s-box or block substitution table. Accordingly, Applicant submits that Mittenthal also fails to teach or suggest combining linear orthomorphisms to generate a non-linear block substitution table.

c. The Combination Of The Luyster And Mittenthal References

Luyster discloses block encryption methods that may use a block substitution table or s-box. *See* Luyster at Abstract. Mittenthal discloses methods of generating s-boxes. *See* Mittenthal at Abstract. Therefore, a combination of the references is a block encryption method according to Luyster that uses s-boxes generated according to Mittenthal. This combination certainly does not teach or suggest combining two linear orthomorphisms to create a non-linear s-box.

4. Neither Luyster Nor Mittenthal, Taken Individually Or Together, Teach Generating A Maximal Non-Linear Substitution Table

a. The Luyster Reference

Applicant submits that the Luyster reference fails to teach or suggest generating a maximal non-linear substitution table. In fact, the Office does not even assert that the Luyster reference teaches generating a maximal non-linear substitution table. *See* Office Action at 5. Further, although the Luyster reference discusses desirable qualities of block substitution tables, or s-boxes, it does not even mention non-linearity, let alone maximal non-linearity, as a desirable quality. *See* Luyster at col. 47, line 45 – col. 48, line 9.

b. The Mittenthal Reference

The Office asserts that Mittenthal discloses, "using nonlinear orthomorphisms to create maximal nonlinear block substitution tables." *See* Office Action at 5. Again, Applicant disagrees. Mittenthal does teach methods of generating s-boxes that are non-linear (*i.e.*, by converting a linear orthomorphism into a non-linear orthomorphism that can be used as an s-box). Nowhere does Mittenthal teach or suggest, though, that the resulting non-linear orthomorphism is maximally non-linear. *See* Mittenthal at Abstract.

c. **The Combination Of The Luyster And Mittenthal References**

As discussed above, the combination of the Luyster and Mittenthal references would be a method of applying a block encryption cipher according to Luyster using an s-box generated according to Mittenthal. Applicant submits that nothing about this combination teaches or suggests generating a maximal non-linear substitution table.

B. **The Section 112 Rejections Are Improper And Should Be Withdrawn**

In the Office Action, the pending claims (claims 1-2, 4, 7-10, 12-13, 16-17 and 20-21) were also rejected under 35 U.S.C. section 112 because the Examiner asserts that they fail to particularly point out and distinctly claim the subject matter which Applicant regards as his invention. *See* Office Action at 4. The basis of the rejection is the term "maximal," which is used in the claims to describe the degree of non-linearity of block substitution tables. The Office Action asserts that the term "maximal" is, "a relative term that renders the claim indefinite." *See id.* Applicant respectfully disagrees.

Determining whether a block substitution table is maximally non-linear is well within the ordinary skill in the art. Methods for calculating the non-linearity of a mapping (*e.g.*, a block substitution table) are well known in the art. For example, the present application describes an exemplary method of doing so. *See* Specification at pp. 3-4, Equations 1-2. It is also well known in the art that there is a maximal value for the non-linearity of a mapping that cannot be exceeded. Applicant submits that methods for determining whether a given mapping is maximally non-linear are well within the ordinary skill in the art. Accordingly, Applicant submits that use of the word "maximal" does particularly point out and distinctly claim the subject matter which Applicant regards as his invention, and therefore respectfully requests that the rejections under 35 U.S.C. section 112, second paragraph be withdrawn.

VIII. CLAIMS APPENDIX

1. A method of deterministically generating maximal nonlinear block substitution tables for a predetermined block size, comprising:

selecting a first generating function;

selecting a second generating function;

selecting first and second sets of complete linearly independent numbers;

calculating first and second linear orthomorphisms from the generating functions and the sets of linearly independent numbers; and

creating maximal nonlinear block substitution tables by combining the linear orthomorphisms, the block substitution tables for use in encrypting clear text messages.

2. The method of claim 1, wherein selecting a first generating function includes selecting a first primitive generating function.

3. (Canceled)

4. The method of claim 1, wherein selecting a second generating function includes selecting a second primitive generating function.

5-6. (Canceled)

7. The method of claim 1, wherein calculating first and second linear orthomorphisms includes calculating first and second maximal linear

orthomorphisms from the generating functions and the sets of linearly independent numbers.

8. The method of claim 1, further comprising rotating the second linear orthomorphism.

9. The method of claim 8, wherein rotating the second linear orthomorphism includes rotating corresponding cycles of the second linear orthomorphism.

10. The method of claim 1, wherein selecting a second generating function includes selecting a second generating function which is a complement of the first generating function.

11. (Canceled)

12. The method of claim 1, wherein selecting first and second sets of linearly independent numbers includes selecting a second set of linearly independent numbers that is identical to the first set of linearly independent numbers.

13. The method of claim 1, wherein selecting first and second sets of linearly independent numbers includes selecting a second set of linearly

independent numbers that is not identical to the first set of linearly independent numbers.

14-15. (Canceled)

16. A computer-implemented method for deterministically generating maximal nonlinear block substitution tables from binary data, comprising:

selecting a first set of a plurality of complete linearly independent numbers from the binary data;

selecting a second set of a plurality of complete linearly independent numbers from the binary data;

generating a plurality of linear orthomorphisms using first and second recursive generating functions and the first and second sets of linearly independent numbers; and

setting the maximal nonlinear substitution tables based on a combination of the linear orthomorphisms, the substitution tables for use in encrypting clear text messages which are in the form of a sequence of binary numbers.

17. The method of claim 16, wherein the second generating function is a complement of the first generating function.

18-19. (Canceled)

20. A system, comprising:

a communications link;

a first computer in communication with the communications link; and

a second computer in communications with the communications link, the second computer having an ordered set of data and instructions stored thereon which, when executed by the second computer, cause the second computer to perform the steps of:

selecting a first generating function;

selecting a second generating function;

selecting first and second sets of complete linearly independent numbers;

calculating first and second linear orthomorphisms from the generating functions and the sets of linearly independent numbers; and

creating maximal nonlinear block substitution tables by combining the linear orthomorphisms, the block substitution tables for use in encrypting clear text messages.

21. A computer-readable medium having stored thereon instructions which, when executed by a processor, cause the processor to perform the steps of:

selecting a first generating function;

selecting a second generating function;

selecting first and second sets of complete linearly independent numbers;

calculating first and second linear orthomorphisms from the generating functions and the sets of linearly independent numbers; and creating maximal nonlinear block substitution tables by combining the linear orthomorphisms, the block substitution tables for use in encrypting clear text messages.

22. (canceled)

IX. EVIDENCE APPENDIX

Not Applicable

X. RELATED PROCEEDINGS APPENDIX

Not Applicable

XI. CONCLUSION

For the foregoing reasons, Applicant submits that the rejections of claims 1-2, 4, 7-10, 12-13, 16-17 and 20-21 in the Office Action are improper and should be reversed.

Respectfully submitted,

Date: 6/2/06

Christopher G. Wolfe
Reg. No. 56264

KIRKPATRICK & LOCKHART NICHOLSON GRAHAM LLP
Henry W. Oliver Building
535 Smithfield Street
Pittsburgh, PA 15222

Ph. (412) 355-6798
Fax (412) 355-6501